REMARKS

Applicant respectfully requests reconsideration of the present application in view of the foregoing amendments and in view of the reasons that follow.

Claims 1, 9, 17, 25, and 33 are currently being amended.

This amendment changes claims in this application. A detailed listing of all claims that are, or were, in the application, irrespective of whether the claim(s) remain under examination in the application, is presented, with an appropriate defined status identifier.

In the Office Action, claims 1-3, 8-11, 16-19, 24-27, 32-35, and 40-43 were rejected under 35 U.S.C. § 103(a) as being obvious over Lewallen (U.S. Patent No. 6,801,224) in view of Stanley ("Extend Office 2000 Applications with Custom-Coded Add-Ins," 8/00).

Claim 1, as amended, recites that a method for dynamically developing a user interface in an existing software application comprises, *inter alia*, invoking a user interface developer component for creating the user interface during the execution of the software application from within the software application, identifying one or more fields to include in the user interface, associating a field type for each of the identified one or more fields, and associating the user interface with a function of the software application.

As disclosed in the Background, Lewallen noted that internally called plug-ins execute within a limited area of execution within a web browser and cannot utilize many of the web browser features outside of their limited area of execution, such as movie and audio player programs embedded as plug-ins in a web browser that cannot be resized or repositioned like the general web browser window (col. 1, lines 55-51). Accordingly, Lewallen discloses that it would be desirable to embed a plug-in into a web browser window to appear integrated with the web browser and allow the plug-in application to implement many of the web browser graphical user interface (GUI) component abilities, such as the capability to resize and reposition the embedded plug-in (col. 1, lines 62-67).

In the Detailed Description, Lewallen discloses that programs 2a, b, c include language statements from different programming languages or protocols, such as Java and

non-Java standard API interfaces, such as the W3C API interfaces (col. 3, lines 12-15). After the mixed statement programs 2a, b, c are written to include Java programming language statements as well as W3C API interface calls, the mixed statement program 2a, b, c is processed by a bridge 4, which maintains an API mapping 8 of W3C API interfaces to the corresponding implementation of the W3C API interface in a user interface (UI) program 10 that implements the Document Object Model (DOM), such as Internet Explorer, Netscape Communicator and Navigator, Mozilla, the Scalable Vector Graphics format used by Adobe Systems, Inc., or any other user interface that implements the DOM (col. 3, lines 53-63).

The user interface program 10 includes user interface (UI) APIs 12 that are used to manipulate user interface (UI) objects 14 that implement the elements and components of the observable user interface features produced by the user interface program 10, and a user interface layout engine 16 transforms the UI APIs 12 and UI objects 14 to the native operating system objects and interfaces 18 on which the browser layout engine 16 was written to operate (col. 3, line 64 – col. 4, line 4). With this approach, the mixed statement programs 2a, b, c can generate a user interface that has the same look-and-feel as the commonly used user interface 10 with which the user is intimately familiar (col. 4, lines 64-67). Because the bridge 4 maps to user interface APIs 12, the mixed statement programs 2a, b, c may execute on any operating system on which the user interface 10 may execute, and the user interface layout engine 16 handles the conversion of the browser APIs 12 and objects 14 to the specific operating system 6 platform (col. 5, lines 1-6).

In addition to allowing mixed statement programs to be displayed on any operating system on which the user interface 10 (e.g., web browser) executes by mapping non-user interface APIs to the user interface APIs 12, Lewallen discloses the ability to create a program or document, such as a web page or user interface, using the non-user interface APIs (col. 7, line 43 – col. 9, line 17). Thus, according to Lewallen, a user can generate web pages and user interfaces for a web browser using APIs different from the APIs of the web browser. Further, the generated web pages and user interfaces can be displayed in the web browser by mapping the different APIs to the APIs of the web browser.

In contrast to claim 1, Lewallen fails to disclose or suggest invoking a user interface developer component for creating the user interface during the execution of the software application from within the software application. In particular, there is nothing in Lewallen that discloses or suggests that the environment for creating the user interface is invoked from within the application on which the user interface is being implemented. Although Lewallen discloses integrating into a browser application the programming for interpreting various types of interfaces, there is no disclosure or suggestion that the application used to create the various types of interfaces is invoked from that browser application. In fact, the structure of Lewallen's Fig. 1, as well as the associated description, specifically suggests that the mixed statement programs 2a-2c are created independently of the operating system 6 upon which the mixed statement programs 2a-2c are transformed and displayed.

Even if combinable, Stanley fails to cure the deficiencies of Lewallen. Like Lewallen, Stanley fails to disclose or suggest invoking a user interface developer component for creating the user interface during the execution of the software application from within the software application as recited in claim 1. Rather, Stanley specifically discloses that a separate application, Visual Basic 6.0, is used to generate interfaces for an application, such as Microsoft Word. Accordingly, even if combinable, claim 1 is patentably distinguishable from the combination of Lewallen and Stanley.

Claims 2, 3, 8, and 41-43 are also patentably distinguishable from the combination of Lewallen and Stanley by virtue of their dependence from claim 1, as well as their additional recitations. Claims 9-11, 16-19, 24-27, 32-35, and 40 are patentably distinguishable from the combination of Lewallen and Stanley for reasons analogous to claim 1.

Claims 4-7, 12-15, 20-23, 28-31, and 36-39 were rejected under 35 U.S.C. § 103(a) as being obvious over the combination of Lewallen and Stanley in view of Menachemi (U.S. Patent App. Pub. No. 2002/0103810). Even if combinable, Menachemi fails to cure the deficiencies of Lewallen and Stanley. Like Lewallen and Stanley, Menachemi fails to disclose or suggest invoking a user interface developer component for creating the user interface during the execution of the software application from within the software application as recited in claim 1. Accordingly, claims 4-7 are patentably distinguishable from

the combination of Lewallen, Stanley, and Menachemi by virtue of their dependence from claim 1, as well as their additional recitations. Claims 12-15, 20-23, 28-31, and 36-39 are patentably distinguishable from the combination of Lewallen, Stanley, and Menachemi for reasons analogous to claim 1.

Applicant believes that the present application is now in condition for allowance. Favorable reconsideration of the application as amended is respectfully requested.

The Examiner is invited to contact the undersigned by telephone if it is felt that a telephone interview would advance the prosecution of the present application.

The Commissioner is hereby authorized to charge any additional fees which may be required regarding this application under 37 C.F.R. §§ 1.16-1.17, or credit any overpayment, to Deposit Account No. 19-0741. Should no proper payment be enclosed herewith, as by a check being in the wrong amount, unsigned, post-dated, otherwise improper or informal or even entirely missing, the Commissioner is authorized to charge the unpaid amount to Deposit Account No. 19-0741. If any extensions of time are needed for timely acceptance of papers submitted herewith, Applicant hereby petitions for such extension under 37 C.F.R. §1.136 and authorizes payment of any such extensions fees to Deposit Account No. 19-0741.

Respectfully submitted,

Date December 27, 2005

By Phillip J. Articola for / William T. Ellis
Reg. No. 38,819
Registration No. 26,874

Marc K. Weinstein
Registration No. 43,250

FOLEY & LARDNER LLP
Customer Number: 22428
Telephone: (202) 672-5485
Facsimile: (202) 672-5399